

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

ABD

1. (Currently amended) A digital data processing system with improved access to information stored on a storage device, said system comprising a plurality of first nodes and a second node coupled to one another over a communications pathway, the second node being coupled to the storage device for determining meta data including block address maps to file data in the storage device, and the first nodes being configured for accessing file data from the storage device using said meta data, wherein said system comprises:
 - at least one first node that caches data including meta data for a file accessed by said first node;
 - a file application on said first node configured to get requested file data by accessing said cached data for the file; and
 - a file notification system that sends a file change notification to said first node indicating changes affecting the cached data, wherein the file application on the first node inspects the change notification to determine whether to get the requested file data directly using said cached data, whereby file accesses may be effected for an extended time with data locally cached at first nodes of the system.
2. (Original) The digital data processing system of claim 1, wherein the file application on said first node determines whether requested file data is subject to a change notification, and if so makes a further determination whether cached data at said first node remains valid for the requested file data.

3. (Currently amended) The digital data processing system of claim 1, wherein the file application on said first node:

a3
(ON)

- i) determines whether requested file data is subject to a change notification; and
- ii) applies the cached meta data to directly mount the storage device to access the requested file when the cached data is not subject to a change notification.

4. (Currently amended) The digital data processing system of claim 2, wherein the file application on said first node further determines whether:

- i) to directly access the file data by applying cached meta data associated with the file to directly mount the storage device, or
- ii) to issue a file request to the second node for valid file access meta data or data.

5. (Original) The digital data processing system of claim 1, wherein the file notification system issues client-specific notifications limited to directories or portions of the file system that are to be accessed by each client.

6. (Original) The digital data processing system of claim 1, wherein the file notification system includes an interface layer with a storage system meta data controller for maintaining or acquiring administrative information pertaining to file size and storage locations.

7. (Original) The digital data processing system of claim 1, wherein the file notification system runs on the second node and interfaces with a file system meta data controller to detect changes in file system storage data, issuing a file change notice in response thereto.

8. (Original) The digital data processing system of claim 1, wherein the file notification system limits number of change notifications for a given file to first n changes that occur, where n is a positive integer.

9. (Original) The digital data processing system of claim 1, wherein the file application on the first node implements a decision algorithm to determine whether to apply cached data for a requested file when the requested file is subject to a change notification.

10. (Original) The digital data processing system of claim 1, wherein the file application on the first node intercepts reads and writes, and issues those directly to the storage device while exchanging messages over the communications pathway to permit coordinate file system management tasks performed by the second node.

11. (Original) The digital data processing system of claim 10, wherein the file system management tasks performed by the second node include publication of change data.

12. (Original) The digital data processing system of claim 1, wherein the file shared access coordination system runs on the second node and interfaces with or includes a file system meta data controller interceding in response to at least a first selected file access request applied thereto by a file application on a first node, and transferring data designated by that request between the first node and the peripheral device in accord with current meta data maintained by the file system pertaining to storage of that data on the storage device such that files may be directly transferred while maintaining file coherence and security.

13. (Currently amended) A digital data processing system, comprising a first node and a second node coupled for communication;

a storage device coupled for communication with at least the first node; and
a cache memory coupled to and associated with the ~~the~~ first node, the cache memory
storing administrative data pertaining to files on the storage device,
the second node notifying the first node of changes to administrative data pertaining files
for which the cache memory stores administrative data.

a3
cont
14. (Original) A digital data processing system according to claim 13, wherein the storage device is any of a disk drive, a "jukebox," other mass storage device or other mapped device.

15. (Original) A digital data processing system according to claim 13, wherein the administrative data stored by the cache includes any of a physical storage map and at least a portion of a directory pertaining to files on the storage device.

16. (Original) A digital data processing system according to claim 13, wherein the digital data processing system comprises a network having a file management system, and a file application on the first node applies administrative data in the cache memory pertaining to a file directly mount the storage device.

17. (Original) A digital data processing system according to claim 13, wherein the digital data processing system comprises a network having a file management system, and a file application applies a notification of a change of administrative data pertaining to a given file by passing a request for that file by the first node to the file management system.

18. (Currently amended) A method of sharing storage access in a digital data processing system having a first node and a second node coupled for communication and a storage device coupled for communication with at least the first node, such method comprising the steps of:

caching in a cache memory coupled to and associated with the first node, administrative data pertaining to files on the storage device; ;

communicating, to the first node, changes to administrative data pertaining to files for which the cache memory stores administrative data; ; and

determining, in the first node, whether to apply said cached data for accessing a file thereby reducing network communications. ;

19. (Currently amended) A digital data processing method for improved access to information stored on a storage device, wherein the system includes a storage device, a plurality of first nodes and a second node communicating over a communications pathway, the second node being coupled to the storage device for determining meta data for accessing file data in the storage device, and the first nodes being configured for accessing file data from the storage device using said meta data, wherein said method is characterized by the steps of:

caching meta data for a file accessed by said first node in a cache memory of said first node; ;

providing a file application on said first node configured to get requested file data utilizing said cached data; ;

storing file change notifications at said first node indicating changes that may affect the cached data; ; and

determining, via said change notifications whether said file application may utilize the cached data for a requested file.

20. (Original) The method of claim 19, wherein the step of storing file change notifications is effected by receiving a change list published by the second node and storing the change list.

21. (Original) The method of claim 19, wherein the file application

i) determines whether requested file data is subject to a change notification, and
ii) applies the cached meta data to directly mount the storage device to access
requested file when the cached data is not subject to a change notification.

22. (Original) The method of claim 19, wherein the file application on said first node
operates

*A3
DRAFT*
i) to directly access the file data by applying cached meta data associated with the file
to directly mount the storage device when said cached data is not subject to a change notification,
or
ii) to issue a file request to the second node when said cached data is subject to a
change notification.

23. (Original) The method of claim 19, wherein the file notifications are client-node-specific
notifications limited to directories or portions of the file system that are to be accessed by each
client.

24. (Original) The method of claim 19, wherein the file notification system runs on the
second node operates with a file system meta data controller to detect changes in file system
storage data and issue file change notifications in response thereto.

25. (Original) The method of claim 24, wherein the second node limits number of change
notifications for a given file to first n changes that occur, where n is a positive integer.

26. (Original) The method of claim 19, wherein the file application on the first node
implements a decision algorithm to determine whether to apply cached data for a requested file.

27. (New) The digital data processing system of claim 1, wherein the file notification system runs on the second node and sends the file change notification to the first node, wherein the first node is registered with the second node for receiving the file change notification, wherein the file change notification includes changes to the data and the meta data.

*A3
Am*

28. (New) The digital data processing system of claim 13, wherein the second node notifies the first node via a file notification system that runs on the second node and sends a file change notification to the first node, wherein the first node is registered with the second node for receiving the file change notification, wherein the file change notification includes changes to data and the administrative data stored in the cache memory.

29. (New) The method of claim 18, wherein a file notification system runs on the second node and sends a file change notification to the first node, wherein the first node is registered with the second node for receiving the file change notification, wherein the file change notification includes changes to file data and the administrative data stored in the cache memory.

30. (New) The method of claim 19, wherein a file notification system runs on the second node and sends a file change notification to one first node, wherein the one first node is registered with the second node for receiving the file change notification, wherein the file change notification includes changes to file data and the meta data.